REMARKS

1. Claim Rejections - 35 U.S.C. §103(a) – Claims 1-20

Claims 1-20 are pending in the present application and were rejected in the Office action dated May 7, 2004 under 35 U.S.C. § 103(a) as being unpatentable over Estakhri et al. (U.S. Patent No. 5,835,935) and in view of Konishi et al. (U.S. Patent No. 5,579,502). Applicant respectfully traverses this rejection. However, in order to provide clarification, claims 1, 10, 17, and 20 have been amended. Claims 1, 10, 17, and 20 are independent claims. Claims 2-9 depend from independent claim 1; claims 11-16 depend from independent claim 10; and claims 18-19 depend from independent claim 17. For brevity, only the bases for the rejection of the independent claims are traversed in detail on the understanding that the dependent claims are also patentably distinct over the prior art, as they depend directly from their respective independent claims. Nevertheless, the dependent claims include additional features that, in combination with those of their respective independent claims, provide further, separate, and independent bases for patentability.

The Examiner states, "Estakhri et al. discloses a memory controller 200 for accessing a memory 212 having a plurality of blocks (clusters) each constituted of a plurality of pages (sectors) based on a host address supplied from a host computer." The Examiner further states, "Estakhri also discloses table 144 holding flag information regarding the sectors within the memory, which is consulted at the time of a read or write access." However, the Examiner admits, "Estakhri does not disclose [1] decision means responsive to a request to write user data issued by the host computer for determining whether progressive data writing for writing user data to a target page designated by the host address is possible; and [2] write means responsive to an affirmative determination by the decision means for writing user data to the target page without performing an inter-block data transfer." Nevertheless, the Examiner states, "Konishi et al. discloses searching a management table for a free block in a memory to write data to." Finally, the Examiner asserted that "[i]t would have been obvious to one of ordinary skill in the art at the time the invention was made to add a 'free block' flag to the table of Estakhri et al. and to integrate the free block table searching procedure of Konishi et al. in order to facilitate the finding of free sectors within the memory when computer updates are being applied to the memory clusters."

The Estakhri et al. and Konishi et al. patents do not teach or suggest each and every element of the claimed invention, either alone or in combination. In this regard, the Estakhri et al. patent does not teach or suggest a memory controller that includes a decision means, responsive to a request to write user data issued by the host computer, for determining whether progressive data writing for writing user data to a target page designated by the host address is possible, "wherein the decision means makes the determination based on start page data which was written to a redundant area."

Specifically, in the Estakhri et al. patent, a flag is disclosed for showing whether or not data is written to a redundant area. Importantly, however, although a flag is disclosed for showing whether or not data is written to a redundant area in the Estakhri et al. patent, this data does not correspond to the start page data in the claimed invention.

In the Konishi et al. patent, the start address of a block of a data area in which data could not normally be written, is written in the management table. This is done in order to manage the relationship between the block of the data area and a block of the remedy area in which data was written. Thus, in the Konishi et al. patent, a start page of a block is written in the management table, while in the claimed invention, the number of a page from which the progressive data writing is to be started is written in the address translation table. Therefore, in the claimed invention, the number of a page from which the progressive data writing is to be started is written in the address translation table in order to manage the page from which the progressive data writing is to be started. Further, in the claimed invention, the number of a page from which the progressive data writing is to be started is obtained based on the start page data 33 written to the redundant area. Such a start page search is neither disclosed nor suggested in the Konishi et al. patent.

Referring to FIG. 11 of the Konishi et al. patent, a start address YYYY of a block X is written in a block Y. Specifically, in FIG. 11 of the Konishi et al. patent, in one embodiment where data cannot normally be written in a block of a data area (e.g., where data cannot normally be written in a block X), the data is written in a block (e.g., a block Y) of a remedy area and a start address of the block of the data area is written in a management table (FIG. 11). This is done in order to manage the relationship between the block of the data area (e.g., the block X) in which data could not normally be written and the block of the remedy area (e.g., the block Y).

For example, the start address YYYY of the block X is written in the block Y. As stated above, in the other cited reference of Estakhri et al., a flag is disclosed for showing whether or not data is written to a redundant area.

In stark contrast, one of the advantages of the claimed invention is that start page data is written to a redundant area (See Figure 5), and the number of the start page is set in an address translation table based on the start page data. In the preferred embodiment of Figure 9, a process for writing start page data in the redundant area 26 is shown. When the flash memory cell 16 is in an erased state, this status indicates that the data stored therein is "1." Therefore, when the flash memory cell 16 is in the erased state, the binary number "1111111111111B" is set in the corresponding logical block address 29, the binary number "11B" is set in the start page flag 32, and the binary number "11111B" is set in the start page data 33. However, as shown in Figure 11, since the corresponding logical block address 29 and the start page data 33 are expressed in terms of decimal numbers (instead of binary numbers), the decimal numbers "8191" and "31" are respectively set, instead of the binary numbers "1111111111111B" and "11111B."

Referring now to the initiation of a data write operation, when data is written up to page #3 based on an external write command supplied from a host computer, the start page data of page #0 is set to the binary number "00011B" and the lower bit of the start page flag of page #0 is changed from "1" to "0." Otherwise stated, the start page flag of page #0 is changed from the binary number "11B" to the binary number "10B."

When progressive data writing is performed from page #3, the number of the page where the data writing is to be next started is set in the start page data of page #3. For example, when the data for two pages is to be written based on an external write command supplied from a host computer, the start page data of page #3 is set to the binary number "00101B" and the lower bit of the start page flag of page #0 is changed from "1" to "0." Otherwise stated, the start page flag of page #3 is changed from the binary number "11B" to the binary number "10B." In another example, when the data for three pages is to be written, the start page data of page #3 is set to the binary number "00110B" and the lower bit of the start page flag of page #0 is changed from "1" to "0." Otherwise stated, the start page flag of page #3 is changed from the binary number "11B" to binary number "10B."

Referring now to Figure 9, "a start page search," which identifies the start page in progressive data writing, is started based on the start page data 33 in the redundant area. In this embodiment, the lower bit of the start page flag 32 of page #0 is first referenced and then the lower bit of the start page flag 32 of page #3, indicated by the start page data 33 of page #0, is referenced. Since the lower bit of the start page flag 32 of page #3 is "0," the lower bit of the start page flag 32 of page #5, indicated by the start page data 33 of the page #3, is referenced next. As a result, since the lower bit of the start page flag 32 of page #5 is "1," the progressive data writing is started from page #5. The details of the "start page search" are described in the specification of the present application from page 19, line 14 to page 20, line 2.

The number of the page from which the progressive data writing is to be started is then written in the address translation table 37 as a start page. Then, when the progressive data writing is performed, the start page written in the address translation table is referenced, and data is written in accordance therewith.

Accordingly, Applicant respectfully submits that the 35 U.S.C. § 103(a) rejection of claims 1-20 as unpatentable over Estakhri et al. in view of Konishi et al. has been overcome.

CONCLUSION

Applicant has made an earnest and bona fide effort to clarify the issues before the Examiner and to place this case in condition for allowance. In view of the foregoing discussions, it is believed clear that the differences between the claimed invention and the prior art are such that the claimed invention is patentably distinct over the prior art. Therefore, consideration and allowance of claims 1-20 is believed to be in order, and an early Notice of Allowance to this effect is respectfully requested. If the Examiner should have any questions concerning the foregoing, the Examiner is invited to telephone the undersigned attorney at (310) 712-8319. The undersigned attorney can normally be reached Monday through Friday from about 9:30 AM to 6:30 PM Pacific Time.

Respectfully submitted,

Dated: 8/4/04

Brooke W. Quist
Reg. No. 45,030
Attorney for Applicant
BROWN RAYSMAN MILLSTEIN
FELDER & STEINER LLP
1880 Century Park East, Suite 711
Los Angeles, CA 90067
(310) 712-8300